Attorney Docket No.: 40101/13901
Ref. No.: 2008.005

REMARKS

I. INTRODUCTION

Claims 1, 2, 4-7 and 10-68 are pending in the present application. Claims 1, 5, 6, 21, 34, 47 and 52 have been amended. Claim 3 has been canceled. No new matter has been added. In view of the above amendments and the following remarks, it is respectfully submitted that all of the pending claims are in condition for allowance.

II. CLAIM REJECTIONS – 35 U.S.C. § 102(b)

Claims 1, 2, 4-7, and 10-68 stand rejected under 35 U.S.C. § 102(b), as being anticipated by U.S. Patent No. 5,136,709 to Shirakabe et al. (Shirakabe). (See 07/02/07 Office Action, p. 5, ¶ 4).

Currently amended claim 10 recites, a "method, comprising: creating a loadable module, the loadable module including a module input and a module output; creating an executable program; and executing the executable program, wherein the executable program performs a method comprising the steps of: *setting up input/output channels by connecting a standard input and a standard output of a running operating system kernel to the module input and the module output*; inserting the loadable module into address space of the running operating system kernel, wherein, once the loadable the module is inserted into the address space, the loadable module begins to execute; and waiting for the loadable module to connect via

kernel/user channels and then connecting those kernel/user channels to the input/output channels." (Emphasis added).

Shirakabe describes a method for generating an operating system by a static link-editor. (See Shirakabe, Abstract) Shirakabe states that when a driver is incorporated into an operating system, a need arises for mutual reference between the driver and a kernel of the operating system for control of the input/output devices. (See Id., col. 1, lines 33-65; Figs. 4 and 8.) According to one type of operation, When a user program issues an input/output request, the kernel calls a driver routine associated with the request. (See Id.) According to another operation, the driver routine calls the kernel routine to effect processing which can be controlled only by the kernel. In addition, the driver routine references data managed by the kernel, such as an address of data and a data length of data associated with the input/output operation.

The Examiner asserts that a user generating an executable program that issues input/output requests, such as in Figs. 4 and 8 of Shirakabe would anticipate the "setting up input/output channels..." as recited in claim 5. (See 07/02/06 Office Action, p. 4, ll. 20-27.) However, the limitations of claim 5 have been amended to more clearly and distinctly describe the process of setting up the input/output channels. Specifically, as recited in claim 5, "the loadable module including a module input and a module output" and the input/output channels are set up "by connecting a standard input and a standard output of a running operating system kernel to the module input and the module output." Accordingly, the Applicants respectfully submitted that the description of Shirakabe is silent on the abovementioned process for setting up input/output channels. As noted by the Examiner, the prior art reference only discloses that the

executable program issues an input/output request by the driver calling either a kernel routine or the kernel calling a driver routine. The description of Shirakabe discloses a linkage procedure in Fig. 4 and a driver definition table in Fig. 8. However neither the referenced prior art, not the description of Shirakabe, teach or suggest the setting up of input/output channels, wherein the channels are set up by "by connecting a standard input and a standard output of a running operating system kernel to the module input and the module output," as recited in claim 5. Accordingly, the rejection of claim 5 should be withdrawn. Because claims 6 and 7 depend from, and, therefore, include all of the limitations of claim 5, it is respectfully submitted that these claims are also allowable for at least the reasons stated above.

Claim 1 recites, *inter alia*, "...the loadable module including a module input and a module output, and wherein the execution library comprises one or more routines for transparently loading the loadable module into the running operating system kernel, passing arguments to the loadable module, and terminating and unloading the loadable module after receiving a termination signal, *the one or more routines of the execution library setting up input/output channels by connecting a standard input and a standard output of the running operating system kernel to the module input and the module output*." (Emphasis added). For the reasons previously discussed with reference to claim 5, it is respectfully submitted that the rejection of claim 1 should be withdrawn. Because claims 2, 4 and 10-20 depend from, and, therefore, include all of the limitations of claim 1, it is respectfully submitted that these claims are also allowable for at least the reasons stated above.

Claim 21 recites, *inter alia*, "...the fourth set of computer instructions includes computer instructions for transparently loading the loadable module into a running operating system kernel, passing arguments to the loadable module, and terminating and unloading the loadable module from the running operating system kernel after receiving a termination signal, *the fourth set further includes at least one routine setting up input/output channels by connecting a standard input and a standard output of the running operating system kernel to the module input and the module output.*" (Emphasis added). For the reasons previously discussed with reference to claim 5, it is respectfully submitted that the rejection of claim 21 should be withdrawn. Because claims 22-33 depend from, and, therefore, include all of the limitations of claim 21, it is respectfully submitted that these claims are also allowable for at least the reasons stated above.

Claim 34 recites, *inter alia*, "...fourth means for transparently loading the loadable module into a running operating system kernel, passing arguments to the loadable module, and terminating and unloading the loadable module from the running operating system kernel after receiving a termination signal, *the fourth means includes at least one routine setting up input/output channels by connecting a standard input and a standard output of the running operating system kernel to the module input and the module output.*" (Emphasis added). For the reasons previously discussed with reference to claim 5, it is respectfully submitted that the rejection of claim 34 should be withdrawn. Because claims 35-46 depend from, and, therefore, include all of the limitations of claim 34, it is respectfully submitted that these claims are also allowable for at least the reasons stated above.

Claim 47 recites, *inter alia*, "*...setting up input/output channels by connecting a standard input and a standard output of the running operating system kernel to the module input and the module output...*" (Emphasis added). For the reasons previously discussed with reference to claim 5, it is respectfully submitted that the rejection of claim 47 should be withdrawn. Because claims 48-51 depend from, and, therefore, include all of the limitations of claim 47, it is respectfully submitted that these claims are also allowable for at least the reasons stated above.

Claim 52 recites, *inter alia*, "*...setting up input/output channels by connecting a standard input and a standard output of the running operating system kernel to the module input and the module output...*" (Emphasis added). For the reasons previously discussed with reference to claim 5, it is respectfully submitted that the rejection of claim 52 should be withdrawn. Because claims 53-68 depend from, and, therefore, include all of the limitations of claim 52, it is respectfully submitted that these claims are also allowable for at least the reasons stated above.

Claims 5-7, and 47-51 stand rejected under 35 U.S.C. § 102(b), as being anticipated by Kempf et al., "Cross-Address Space Dynamic Linking," September 1992 (Kempf). (See 07/02/07 Office Action, p. 16, ¶ 5).

Kempf describes a method of dynamic linking to allow a user process to link a program in another address space within compromising the security of the other address space and without requiring the linking process to enter kernel mode. (See Kempf, Abstract). As

noted by the Examiner, the linking process maps a code table from an obtained target address space into the linking address space. (See Id., § 5, p. 7, and Fig. 2). The code table contains a list of the base addresses and sizes of code files that have already been linked. (See Id.). However, similar to Shirakabe, Kempf is silent on setting up input/output channels. Specifically, Kempf fails to teach or suggest "setting up input/output channels by connecting a standard input and a standard output of a running operating system kernel to the module input and the module output," as recited in claim 5. Accordingly, the rejection of claim 5 should be withdrawn. Because claims 6 and 7 depend from, and, therefore, include all of the limitations of claim 5, it is respectfully submitted that these claims are also allowable for at least the reasons stated above.

As stated above, claim 47 recites, inter alia, "...setting up input/output channels by connecting a standard input and a standard output of the running operating system kernel to the module input and the module output..." (Emphasis added). For the reasons previously discussed with reference to claim 5, it is respectfully submitted that the rejection of claim 47 should be withdrawn. Because claims 48-51 depend from, and, therefore, include all of the limitations of claim 47, it is respectfully submitted that these claims are also allowable for at least the reasons stated above.

Attorney Docket No.: 40101/13901
Ref. No.: 2008.005

CONCLUSION

It is therefore respectfully submitted that all of the presently pending claims are allowable. All issues raised by the Examiner having been addressed, an early and favorable action on the merits is earnestly solicited.

Respectfully submitted,

Dated: January 2, 2008

By: _____
Michael J. Marcin (Reg. No. 48,198)

Fay Kaplun & Marcin, LLP
150 Broadway, Suite 702
New York, New York 10038
Tel.: (212) 619-6000
Fax: (212) 619-0276